A dark, atmospheric night scene featuring a person standing on a path, looking up at a bright light source, possibly a street lamp or a celestial body, through the branches of trees.

# Understanding Probabilistic Data Structures with 112,092 UFO Sightings



**redis**labs  
HOME OF REDIS

# Guy Royse

## Developer Advocate

## Redis Labs

 @guyroyse

 [github.com/guyroyse](https://github.com/guyroyse)

 [guy.dev](https://guy.dev)

112,092

Key	Value
summary	Three saucer shaped ships. High in the sky, metallic but no shine. Rather dull gray. They hovered overhead in a V formation.
city	Salem
state	Oregon
date_time	1950-09-15T14:00:00-07:00
shape	disk
duration	15 minutes
report_link	<a href="http://www.nuforc.org/webreports/135/S135871.html">http://www.nuforc.org/webreports/135/S135871.html</a>
text	Three saucer shaped ships. High in the sky, metallic but no shine. Rather dull gray. They hovered overhead in a V formation and from time to time seemed to dip down and they had a bubble shaped top. (NUFORC Note: Witness indicates that the date above is approximate. PD)
city_latitude	44.941247110675775
city_longitude	-123.00423516160726

What's a  
Probabilistic Data  
Structure?



**Guy Royse**  
@guyroyse



## Quick developer survey. Have you heard of probabilistic data structures?

Never heard of 'em.

**58%**

I've heard the term.

22%

I know what they are.

9%

I've used them.

11%

119 votes · Final results

5:34 PM · Dec 12, 2019 · Twitter Web App

# Deterministic Data Structures

Lists



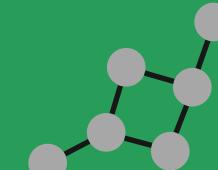
Sets



Arrays



Graphs



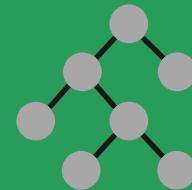
Tuples



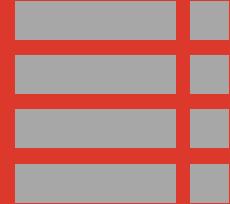
Stacks



Trees



Tables



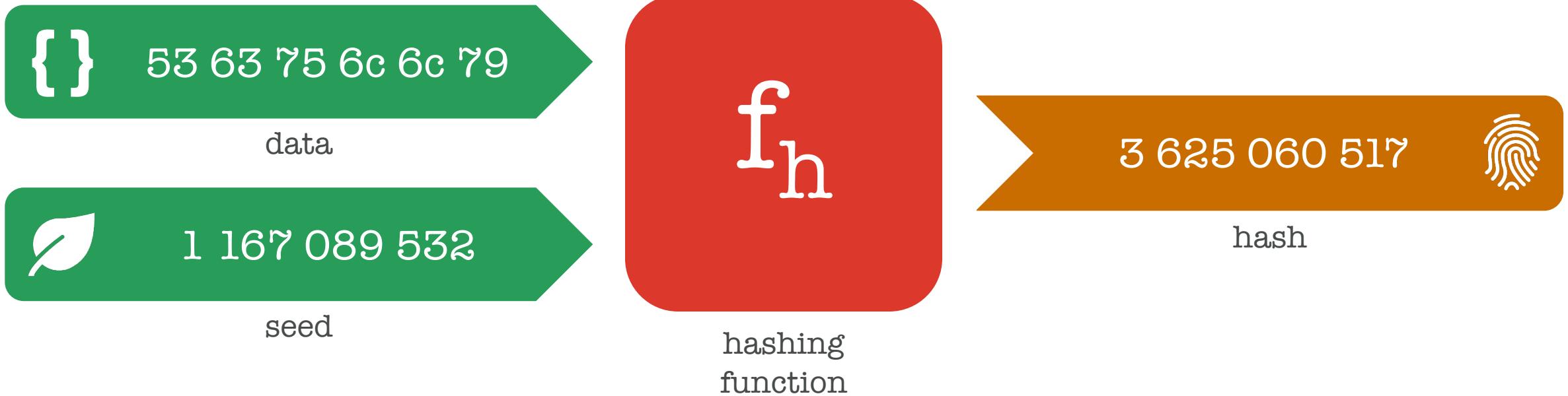
Queues



# Hash Table

Key	Value
city	Salem
state	Oregon
date_time	1950-09-15T14:00:00-07:00
shape	disk
summary	Three saucer shaped ships. High in the sky, metallic but no shine. Rather dull gray. They hovered overhead in a V formation.

# Hashing Functions



# Adding to a Hash Table



Index	Bucket
0	
1	"Salem"
2	
3	
4	
5	
6	
7	
8	
9	

# Adding to a Hash Table



Index	Bucket
0	
1	"Salem"
2	
3	
4	"Oregon"
5	
6	
7	
8	
9	

# Hash Collisions



Index	Bucket
0	
1	"Salem", "disk"
2	
3	
4	"Oregon"
5	
6	
7	
8	
9	



# Reading from a Hash Table



Index	Bucket
0	
1	“Salem”, “disk”
2	
3	
4	“Oregon”
5	“1950-09-15T14:00:00-07:00”
6	
7	“Three saucer shaped ships...”
8	
9	

# Time & Space Complexity



**Time**

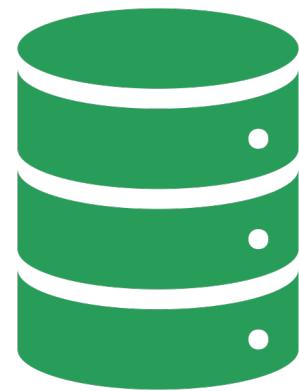


**Space**

# Time & Space Complexity



**Time**



**Space**

# Accuracy Complexity



Time



Accuracy



Space

# Hash Table without Chaining (Probably a Bad Idea)



Index	Bucket
0	
1	"Salem"
2	
3	
4	"Oregon"
5	"1950-09-15T14:00:00-07:00"
6	
7	"Three saucer shaped ships..."
8	
9	

# Hash Collisions (Definitely a Bad Idea)



Index	Bucket
0	
1	"disk"
2	
3	
4	"Oregon"
5	"1950-09-15T14:00:00-07:00"
6	
7	"Three saucer shaped ships..."
8	
9	

# Reading from It (OMG! What Have I done?)

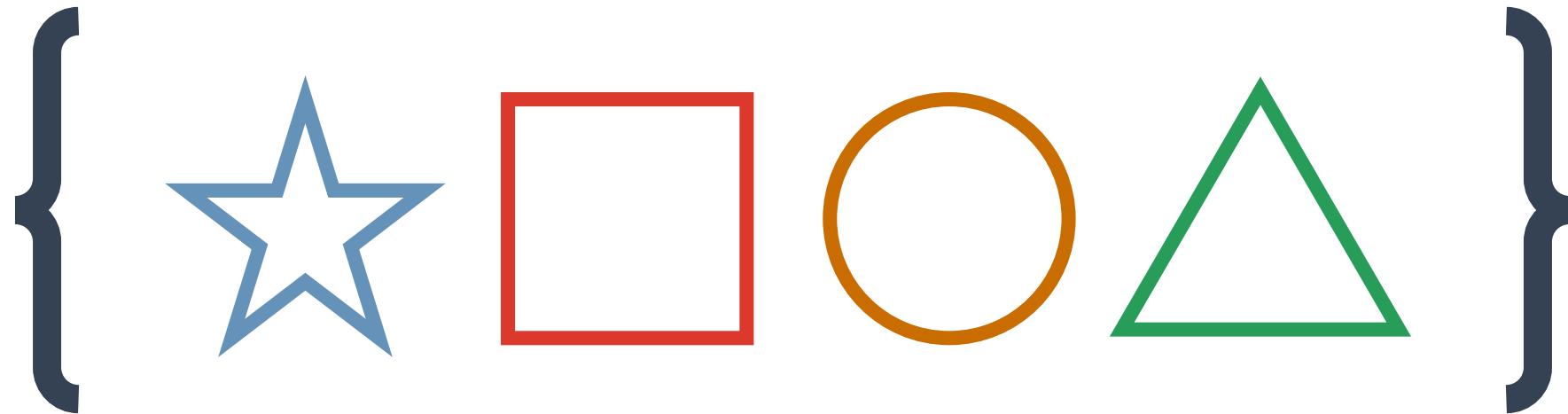


Index	Bucket
0	
1	"disk"
2	
3	
4	"Oregon"
5	"1950-09-15T14:00:00-07:00"
6	
7	"Three saucer shaped ships..."
8	
9	

# A Little Bit Lossy But Good Enough



**Many probabilistic data structures can be  
thought of as peculiar sets.**



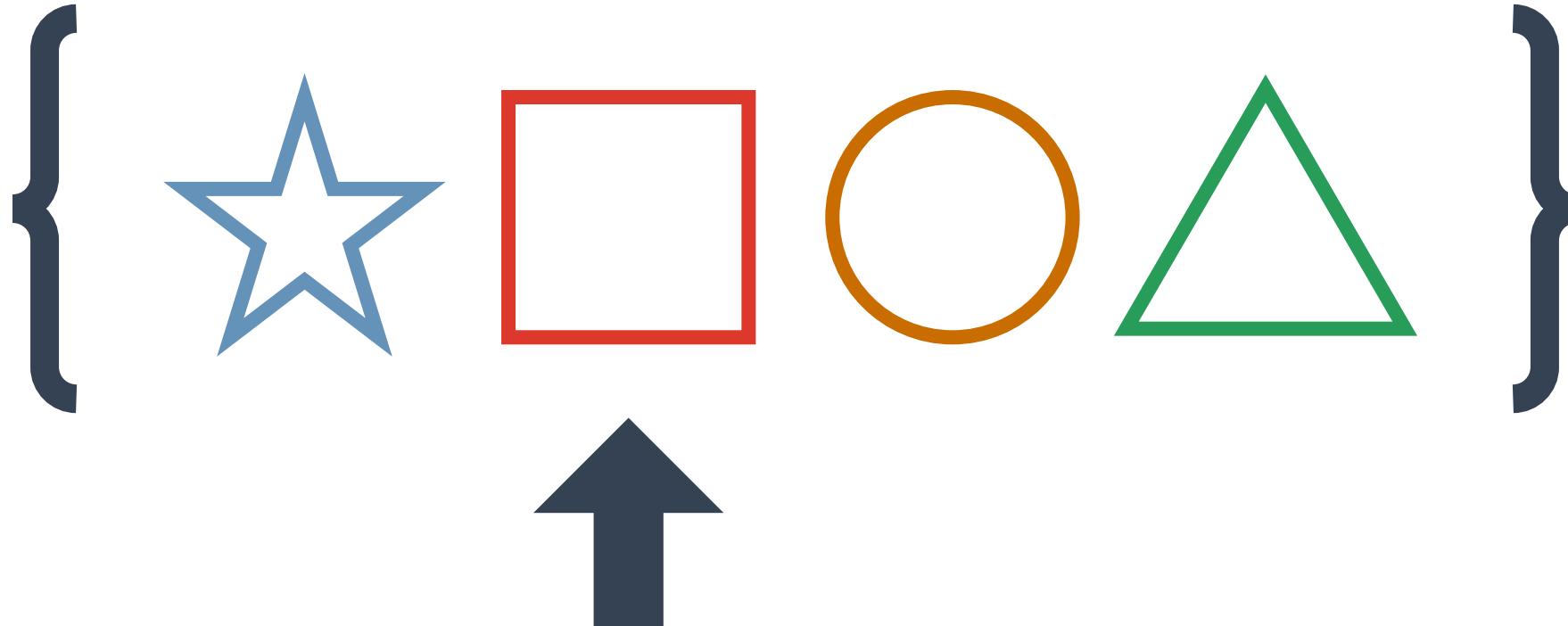
Add · Remove · Get · Union · Intersect · Difference · Membership · Cardinality

# Membership



Is the thing  
in the set?

**A member is any one of the distinct  
objects that make up a set.**

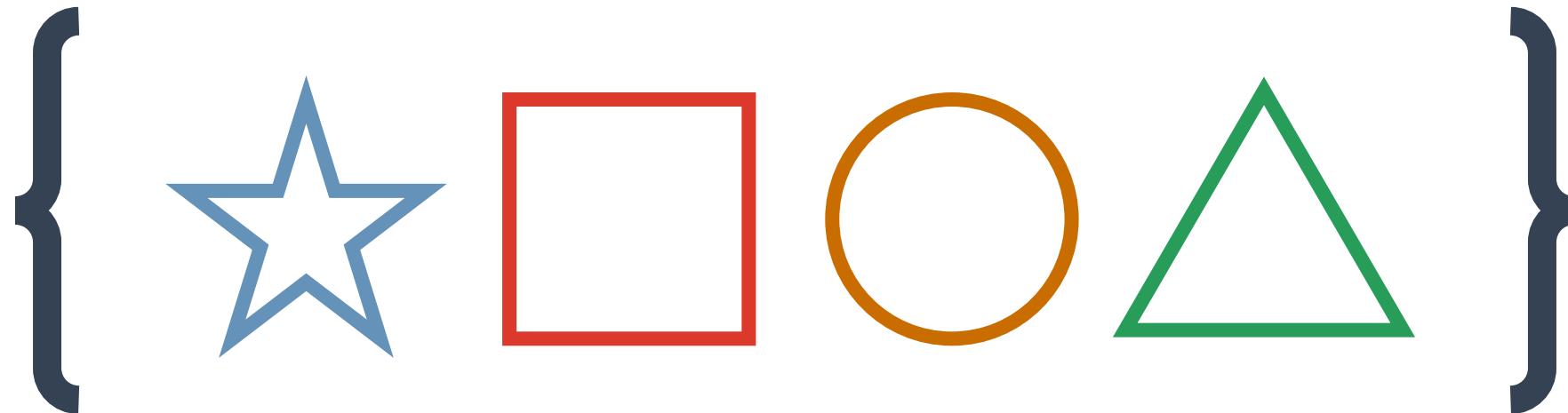


# Cardinality



How many  
of the thing  
are in  
the set?

The number of members in a set.



1

2

3

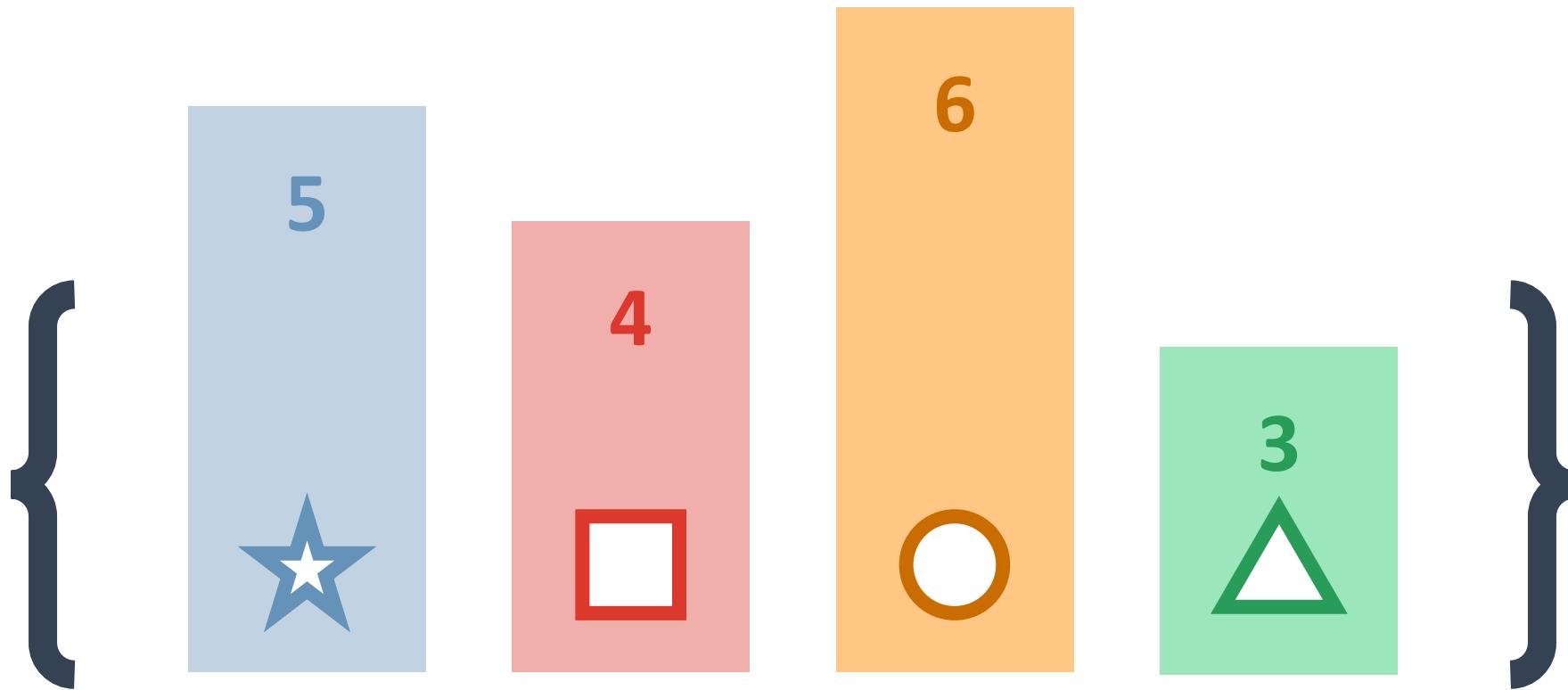
4

# Frequency



How many of  
each thing is in  
the set?

**The number of times a member is found in a set.**

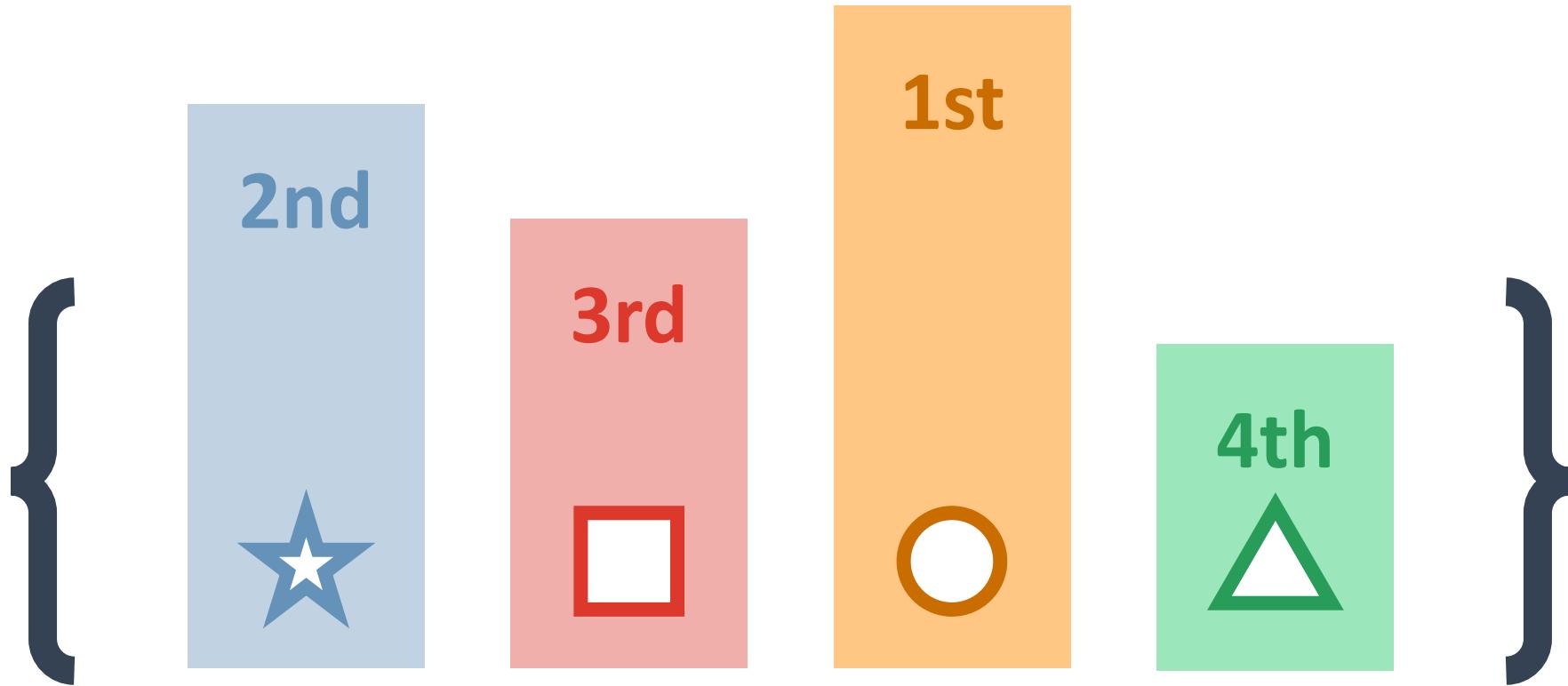


# Rank



Where does  
the thing  
belong  
in the set?

The position of a member of a set relative to other members.

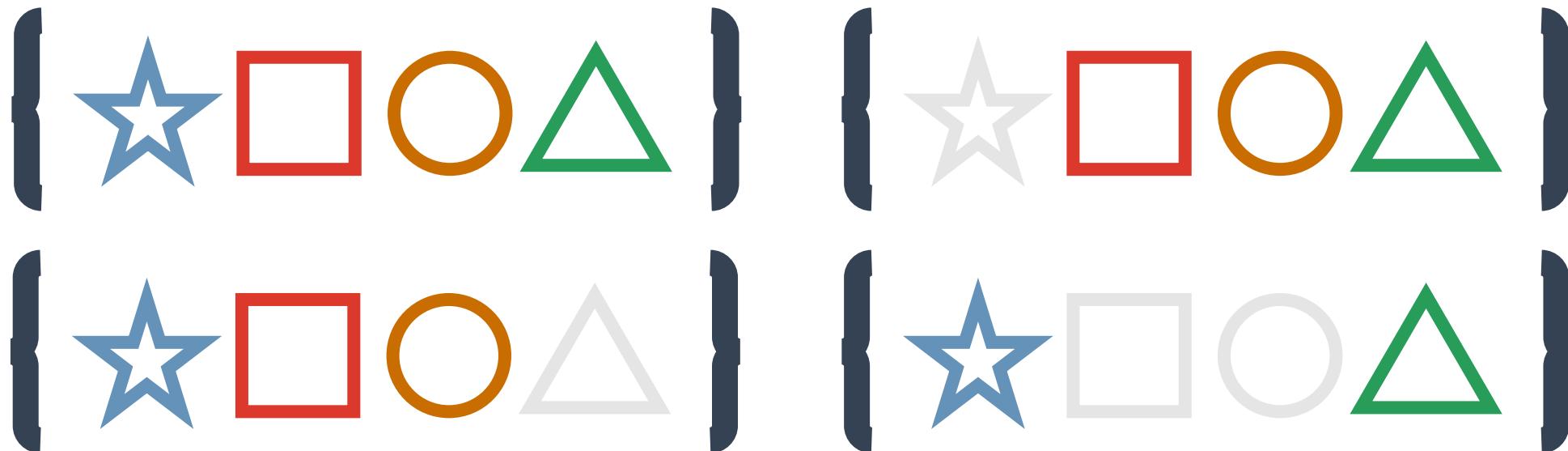


# Similarity



Is the thing  
like the other  
things?

## A measure of the differences between sets.

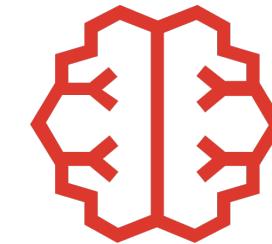
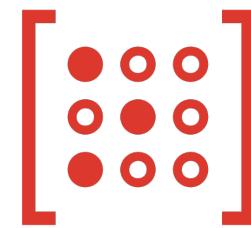
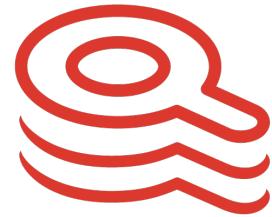
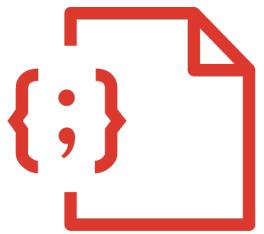
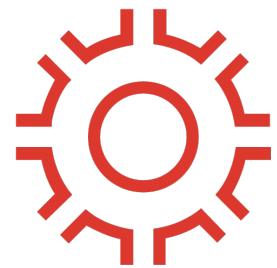


# Lots of Choices

Name	Usage
Bloom Filter <sup>†</sup>	Membership
Cuckoo Filter <sup>†</sup>	Membership
HyperLogLog <sup>‡</sup>	Cardinality
Count Sketch	Frequency
Count-Min Sketch <sup>†</sup>	Frequency
Q-digest	Rank
T-digest	Rank
Heavy Keeper <sup>†</sup>	Rank
Heavy Guardians	Rank
MinHash	Similarity
SimHash	Similarity

<sup>‡</sup> included with Redis <sup>†</sup> included with RedisBloom

# Redis Modules



Bloom Filter\*

# What's a Bloom Filter?



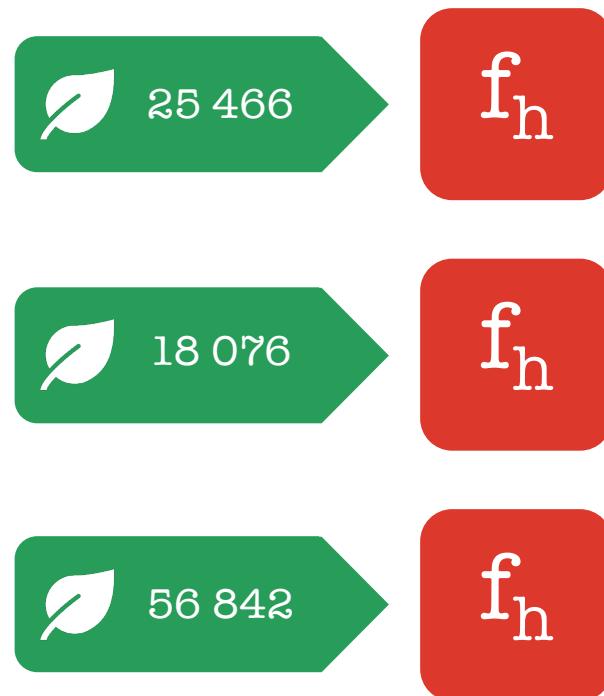
Membership:  
No or Probably

Fixed Size

Fast

# Parts of a Bloom Filter

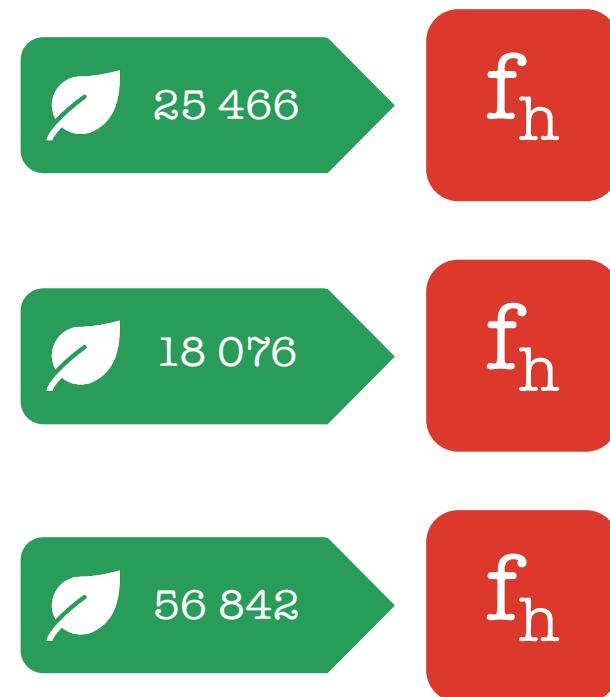
Bit	0	0	0	0	0	0	0	0	0	0
Index	0	1	2	3	4	5	6	7	8	9



# Parts of a Bloom Filter

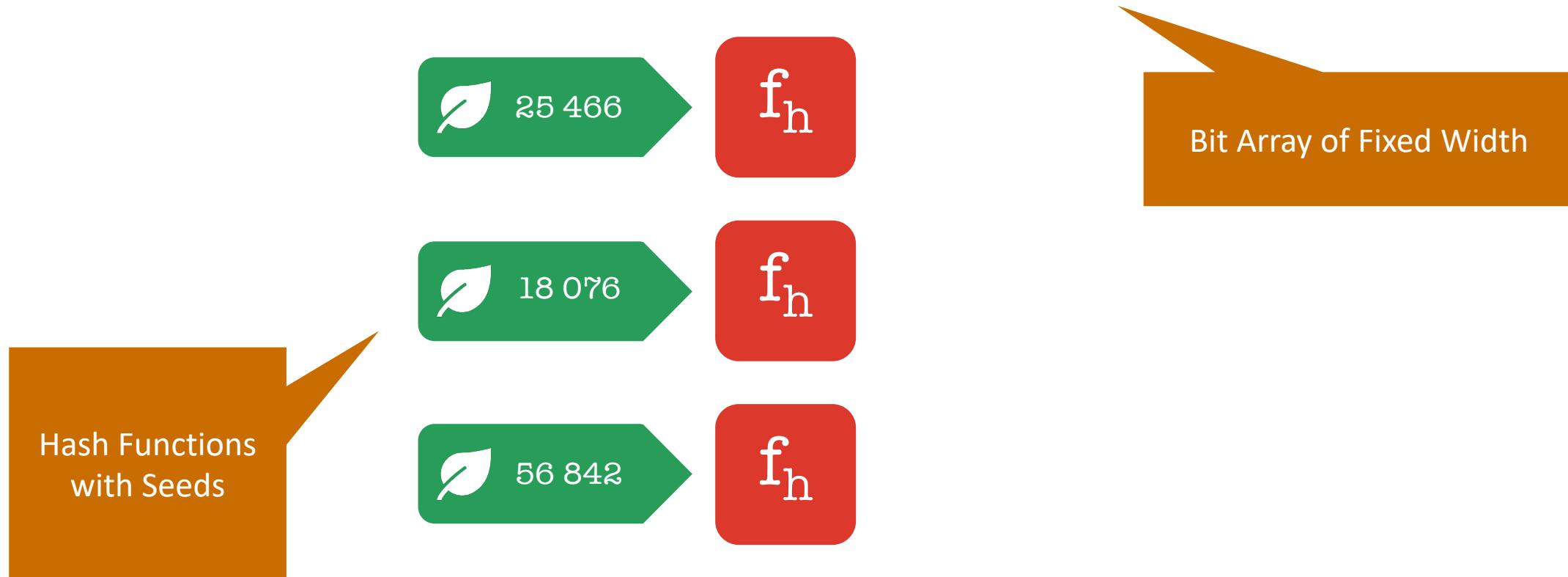
Bit	0	0	0	0	0	0	0	0	0	0
Index	0	1	2	3	4	5	6	7	8	9

Hash Functions  
with Seeds



# Parts of a Bloom Filter

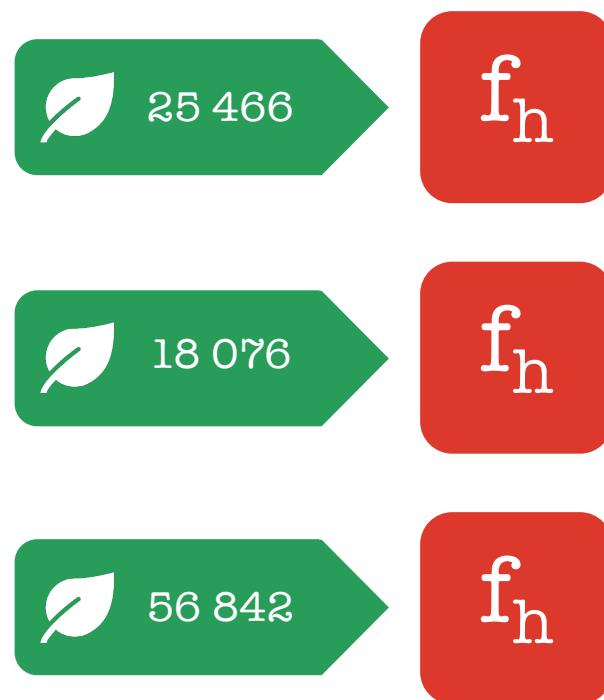
Bit	0	0	0	0	0	0	0	0	0	0
Index	0	1	2	3	4	5	6	7	8	9



# Adding to a Bloom Filter

Bit	0	0	0	0	0	0	0	0	0	0
Index	0	1	2	3	4	5	6	7	8	9

{ } “Megatron  
in the  
bushes”



# Adding to a Bloom Filter

Bit	0	0	0	0	0	0	0	0	0	0
Index	0	1	2	3	4	5	6	7	8	9

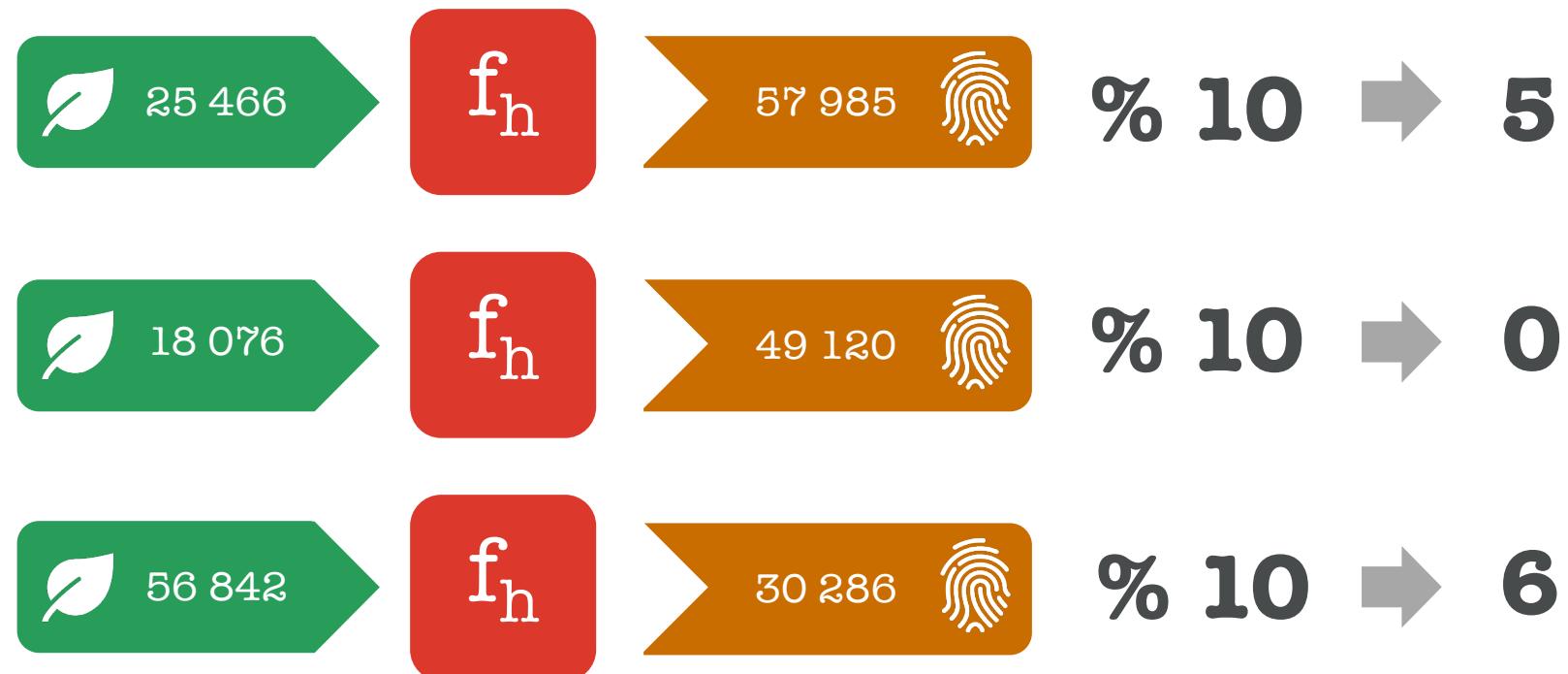
{ } “Megatron  
in the  
bushes”



# Adding to a Bloom Filter

Bit	0	0	0	0	0	0	0	0	0	0
Index	0	1	2	3	4	5	6	7	8	9

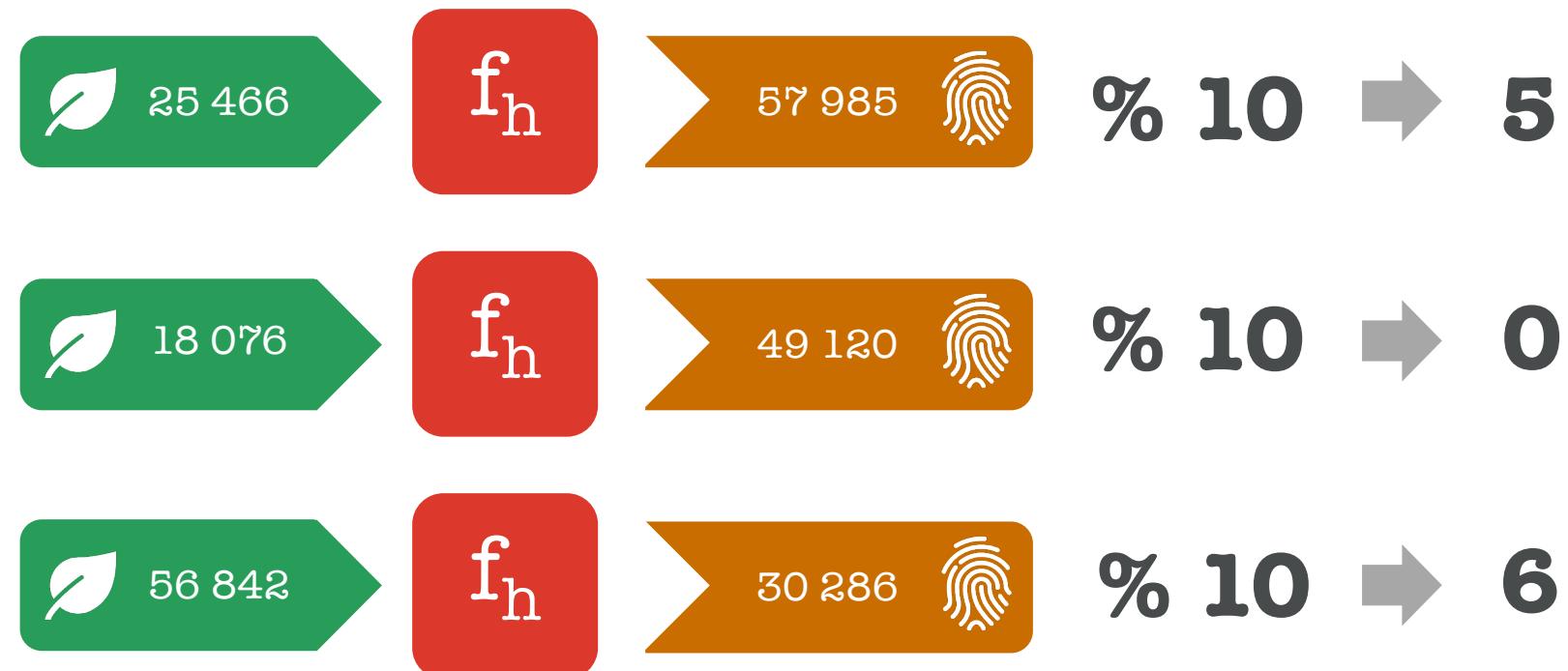
{ } “Megatron  
in the  
bushes”



# Adding to and Reading from Bloom Filter

Bit	1	0	0	0	0	1	1	0	0	0
Index	0	1	2	3	4	5	6	7	8	9

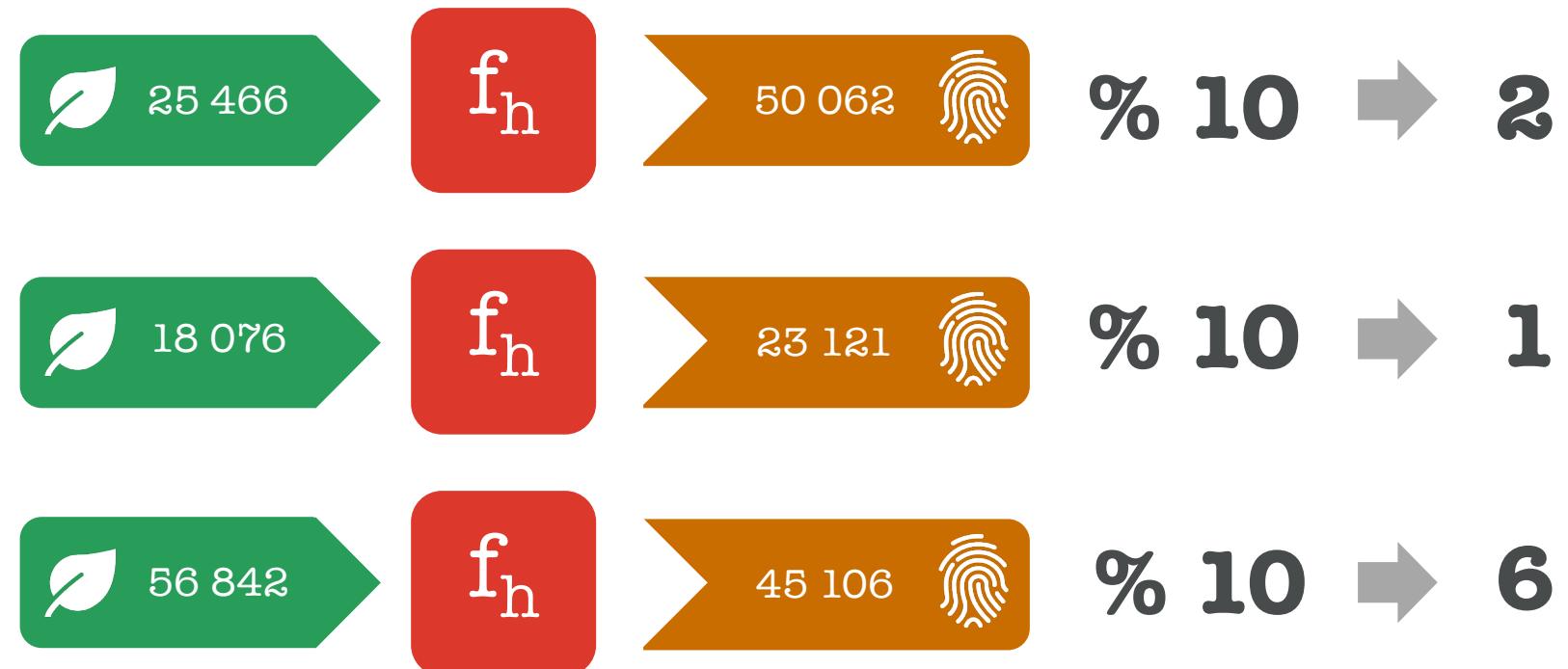
{ } “Megatron  
in the  
bushes”



# Adding to and Reading from a Bloom Filter with Collision

Bit	1	1	1	0	0	1	1	0	0	0
Index	0	1	2	3	4	5	6	7	8	9

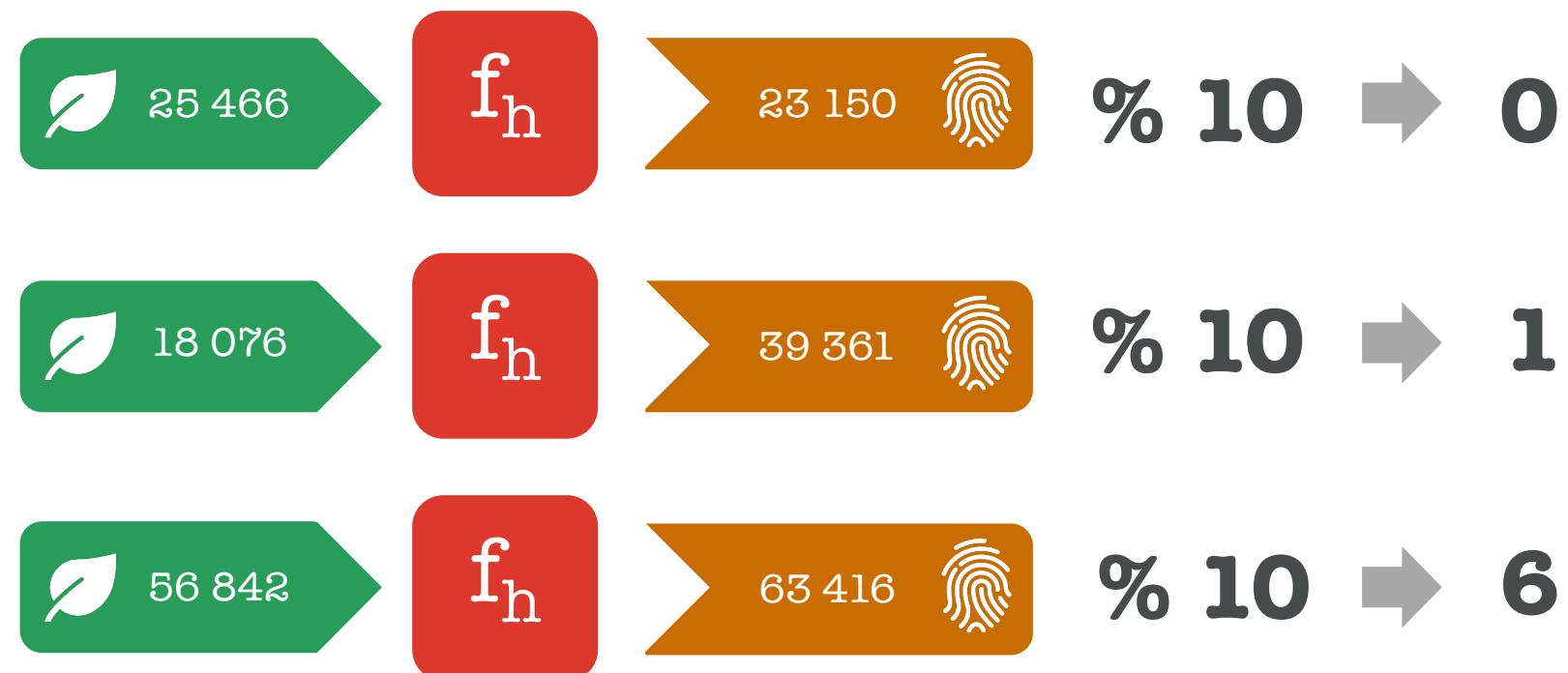
{ } "I SEEN A  
UFO WHEN  
I WAS  
ABOUT 13  
YEARS  
OLD"



# Adding to and Reading from a Bloom Filter with Total Collision

Bit	1	1	1	0	0	1	1	0	0	0
Index	0	1	2	3	4	5	6	7	8	9

{ } “Investig-  
ators from  
the BFRO  
observed a  
glowing  
green/blue  
sphere”



# Computing Hashes & Width

$$p = \left(1 - e^{-kn/mk}\right)^k$$

$$k = (m/n) \cdot \ln(2)$$

- $n \rightarrow$  number of items
- $m \rightarrow$  number of bits
- $k \rightarrow$  number of hashes
- $p \rightarrow$  false positive rate

# Using Bloom Filters in Redis

>	BF.RESERVE	ufo:shapes	0.005	100
	command	key	Error rate	Capacity
OK				

# Adding to Bloom Filters in Redis

>	BF.ADD	ufo:shapes	disk
	command	key	Item to add
(integer) 1			

>	BF.ADD	ufo:shapes	light
	command	key	Item to add
(integer) 1			

>	BF.MADD	ufo:shapes	light	teardrop
	command	key	Item to add	Another item to add
(integer) 0				
(integer) 1				

# Using a Bloom Filters in Redis

>	BF.EXISTS	ufo:shapes	disk
	command	key	Item to add
(integer) 1			

>	BF.EXISTS	ufo:shapes	light
	command	key	Item to add
(integer) 1			

>	BF.MEXISTS	ufo:shapes	light	triangle
	command	key	Item to add	Another item to add
(integer) 1				
(integer) 0				

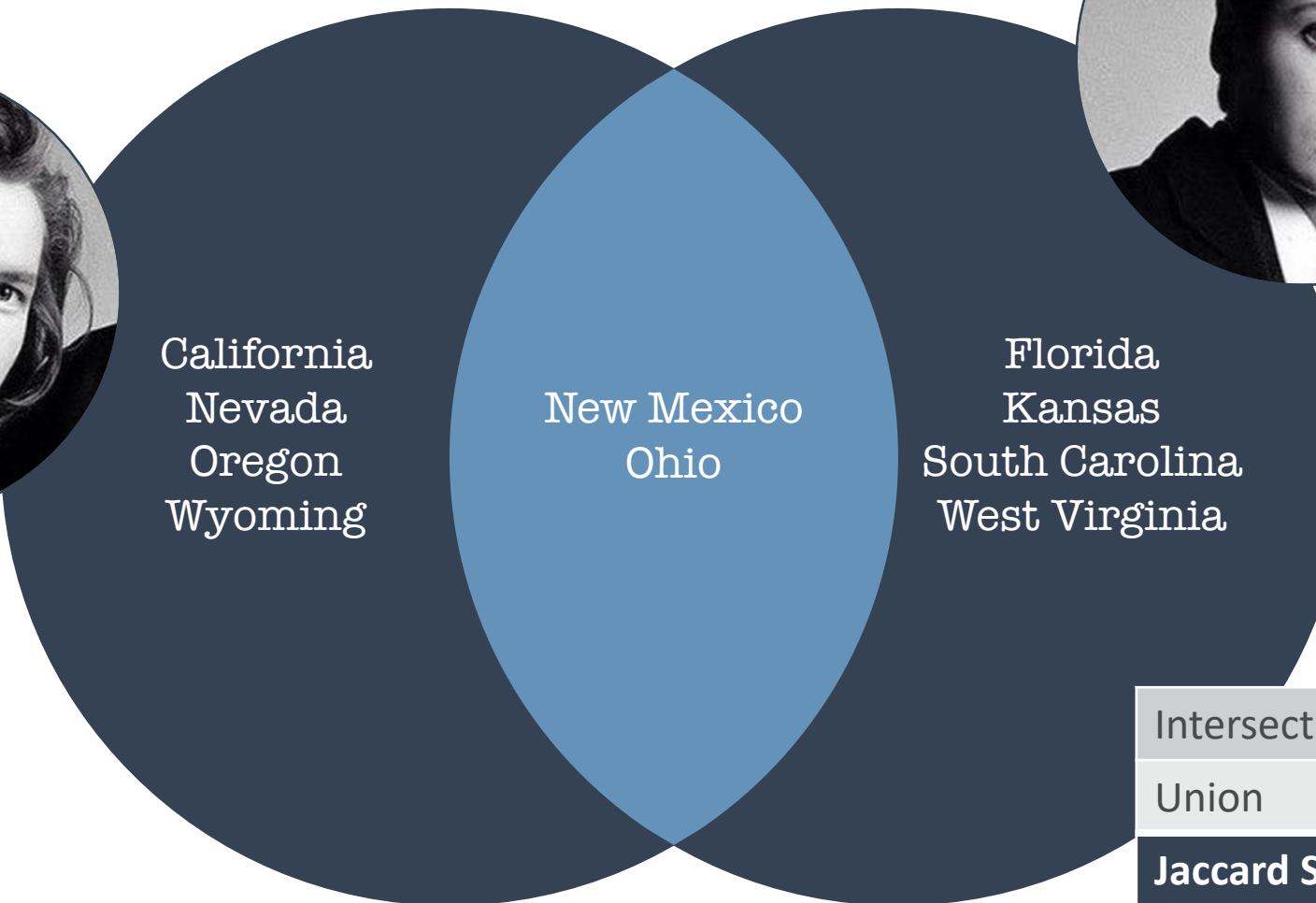
# MinHash

# What's a MinHash?



Determines  
Set Similarity  
Between  
Documents

# Jaccard Similarity



Intersection	2
Union	10
<b>Jaccard Similarity</b>	<b><math>2 / 10 = 0.2</math></b>

# Shingling: Turning Documents into Sets

We noticed to our left,  
farther off into the  
mountains, a bright  
glow.



## Shingles

we noticed to  
noticed to our  
to our left  
our left farther  
left farther off  
farther off into  
off into the  
into the mountains  
the mountains a  
mountains a bright  
a bright glow

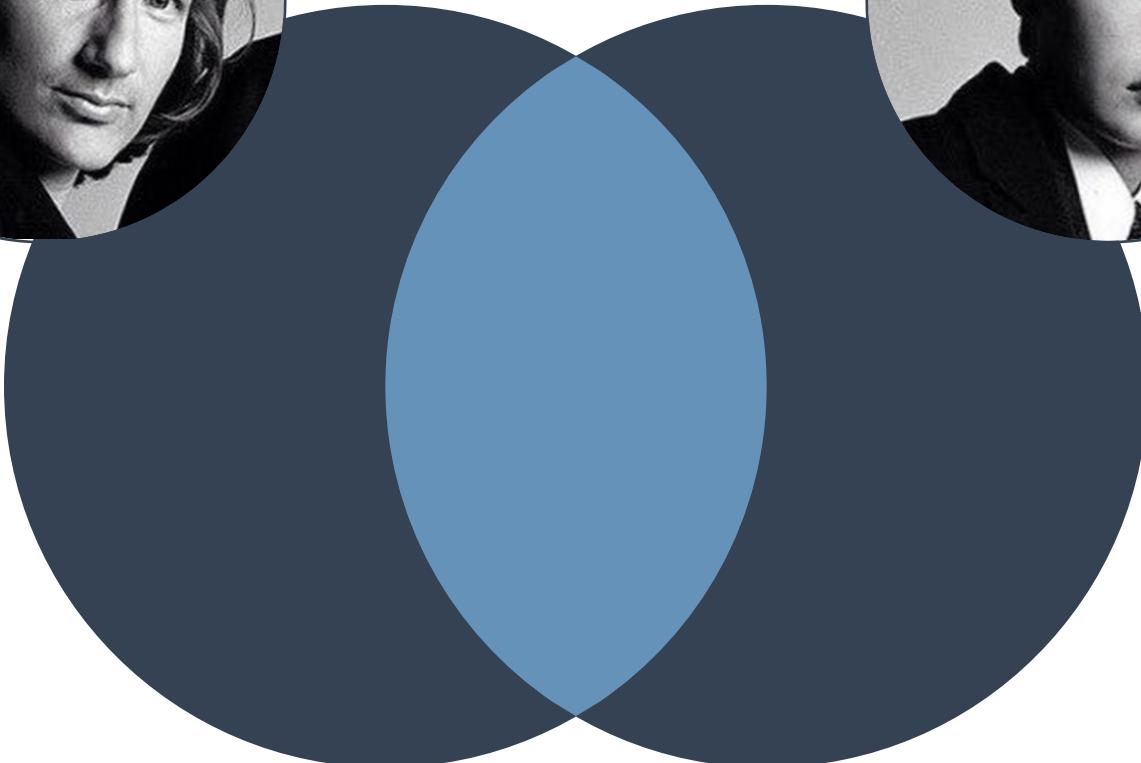
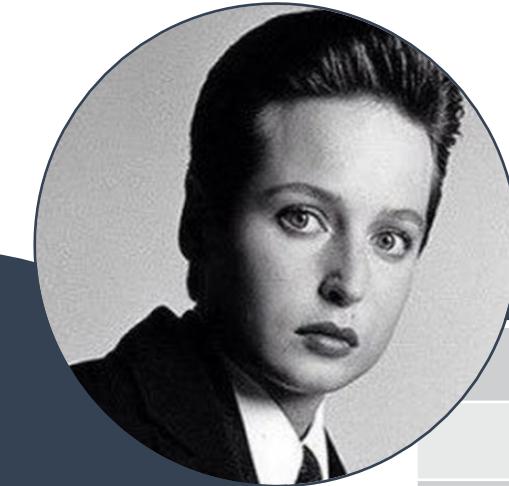
## Sculder's Report

we noticed to  
noticed to our  
to our left  
our left farther  
left farther off  
farther off into  
off into the  
into the mountains  
the mountains a



## Mully's Report

to our left  
our left farther  
left farther off  
farther off into  
off into the  
into the mountains  
the mountains a  
mountains a bright  
a bright glow



Intersection	7
Union	11
Jaccard Similarity	$7 / 11 = 0.636$

# Calculating Sculder's MinHash



## Sculder's Report

we noticed to  
noticed to our  
to our left  
our left farther  
left farther off  
farther off into  
off into the  
into the mountains  
the mountains a



# Calculating Mully's MinHash

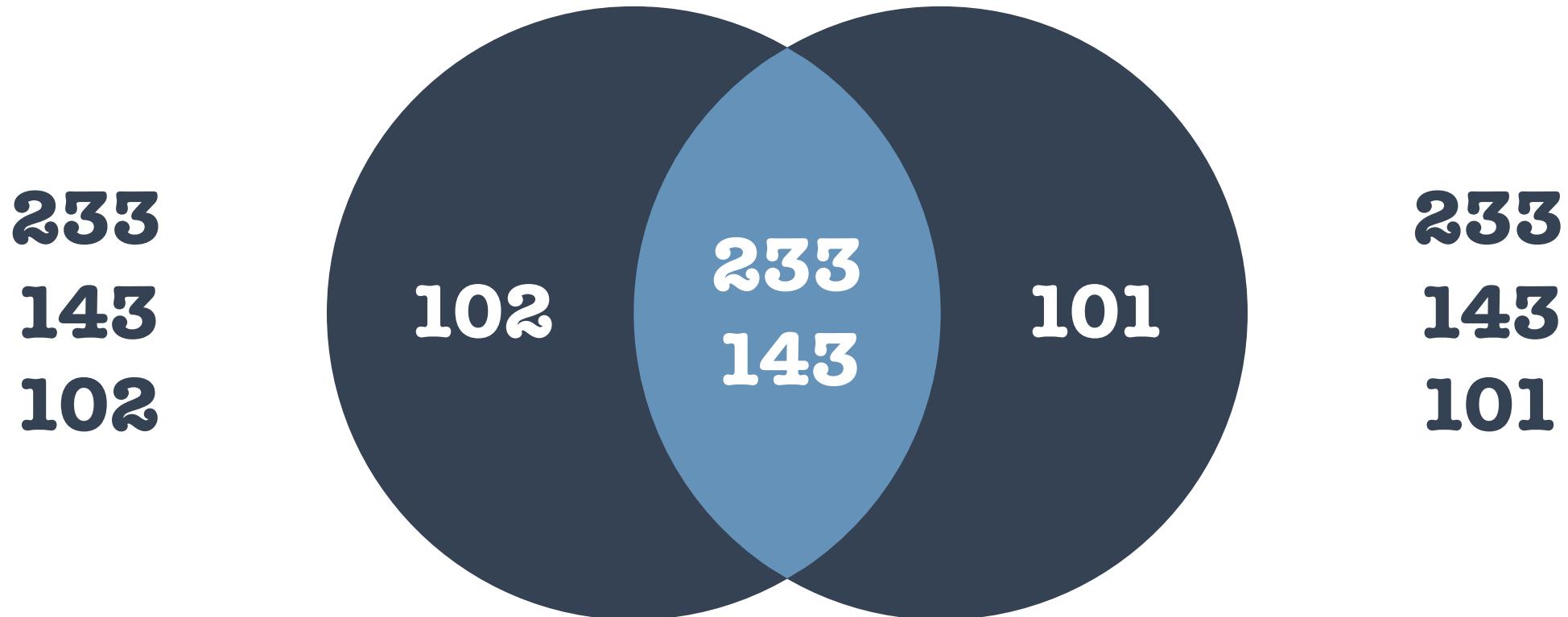


## Mully's Report

to our left  
our left farther  
left farther off  
farther off into  
off into the  
into the mountains  
the mountains a  
mountains a bright  
a bright glow



# Jaccard Similarity of the Minimum Hashes



Intersection	2
Union	4
<b>Jaccard Similarity</b>	<b><math>2 / 4 = 0.5</math></b>

# Comparing the Results

0.636 vs 0.500

TopK\*

(actually Heavy Keeper)

# What's a TopK Heavy Keeper?

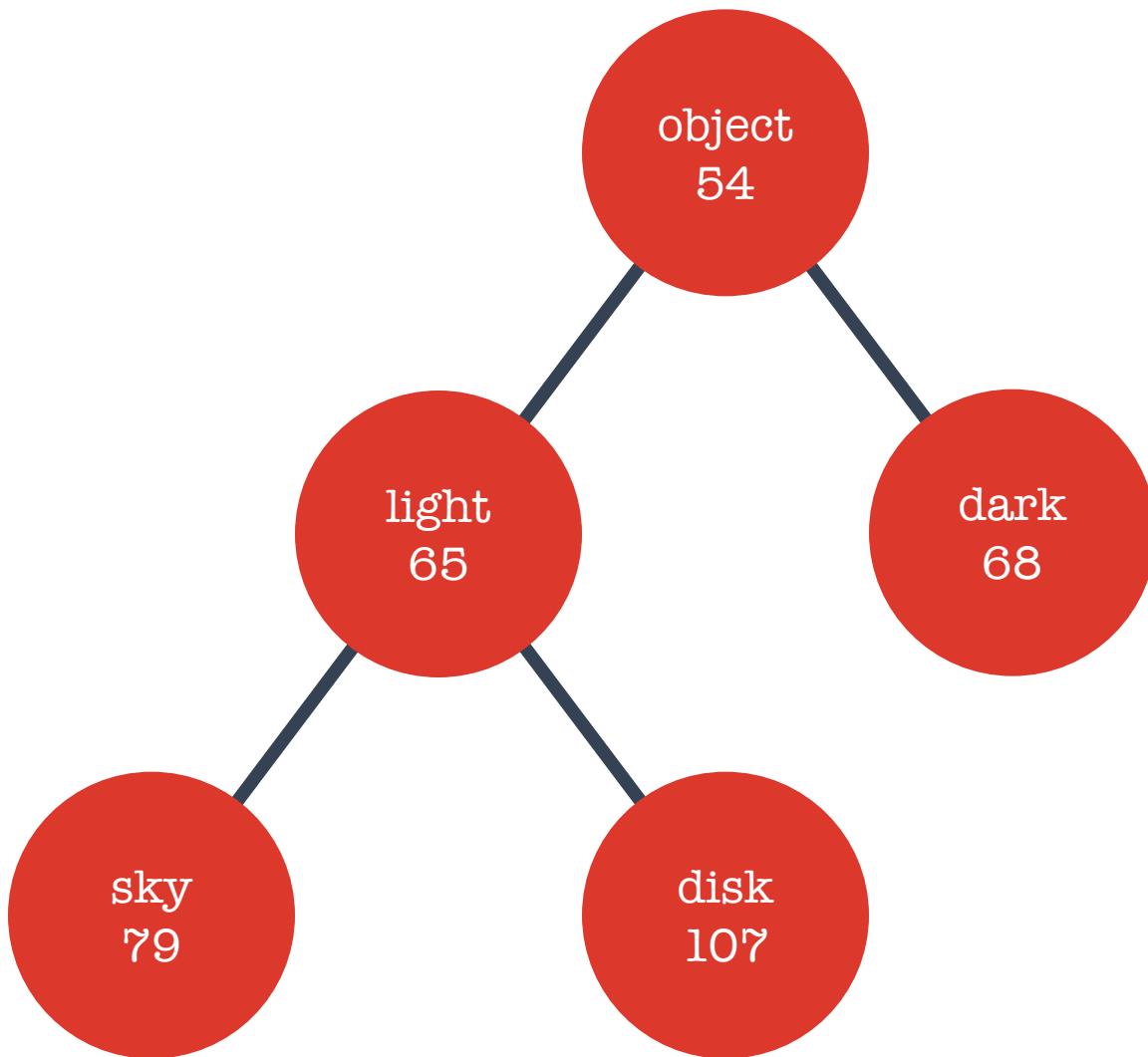


Ranks  
Top k  
Items

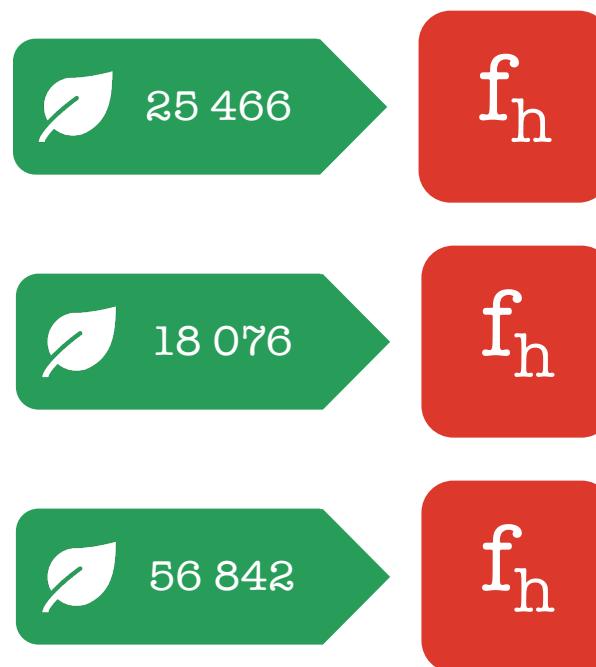
# Two Parts



# MinHeap



# Parts of a Heavy Keeper



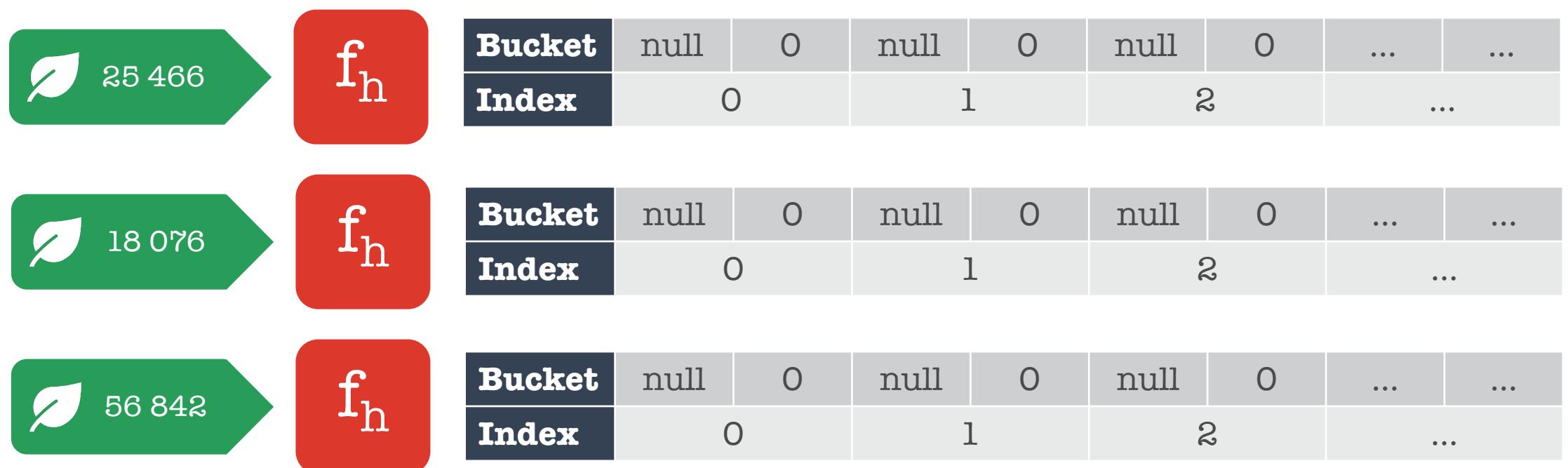
Hash	Count	Hash	Count	Hash	Count	Hash	Count
------	-------	------	-------	------	-------	------	-------

Bucket	1983	15	null	0	1234	3	...	...
Index	0		1		2		...	...

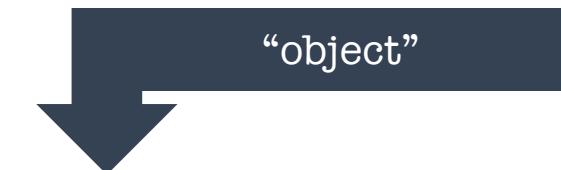
Bucket	null	0	6765	5	3290	11	...	...
Index	0		1		2		...	...

Bucket	9181	0	null	0	null	0	...	...
Index	0		1		2		...	...

# Brand New Heavy Keeper



# Adding a Value



Hash	Count	Hash	Count	Hash	Count	Hash	Count
------	-------	------	-------	------	-------	------	-------

Bucket	null	0	null	0	null	0	...	...
Index	0		1		2		...	...

Bucket	null	0	null	0	null	0	...	...
Index	0		1		2		...	...

Bucket	null	0	null	0	null	0	...	...
Index	0		1		2		...	...

# Adding a Value



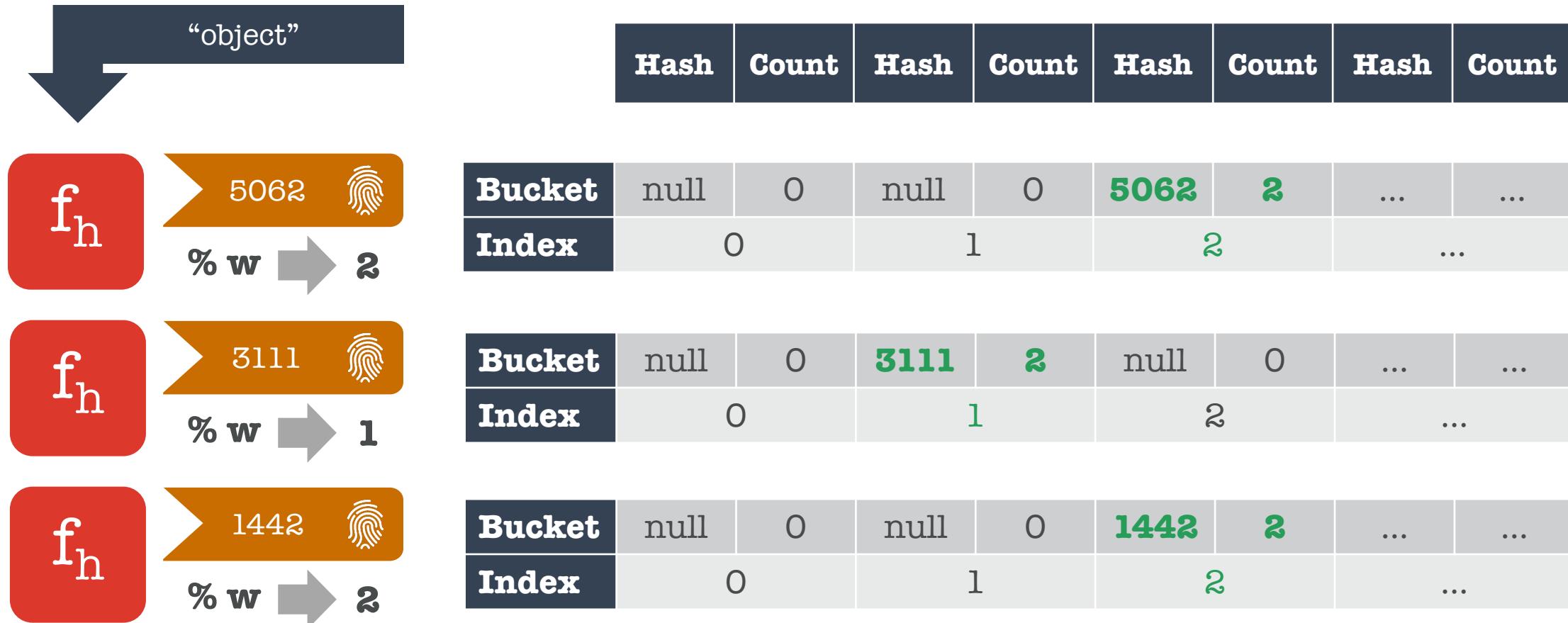
Hash	Count	Hash	Count	Hash	Count	Hash	Count
------	-------	------	-------	------	-------	------	-------

Bucket	null	0	null	0	<b>5062</b>	<b>1</b>	...	...
Index	0		1		<b>2</b>		...	...

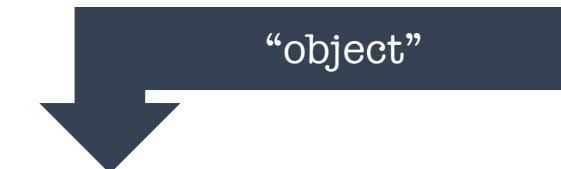
Bucket	null	0	<b>3111</b>	<b>1</b>	null	0	...	...
Index	0		<b>1</b>		<b>2</b>		...	...

Bucket	null	0	null	0	<b>1442</b>	<b>1</b>	...	...
Index	0		1		<b>2</b>		...	...

# Adding the Same Value Again



# Querying the Count



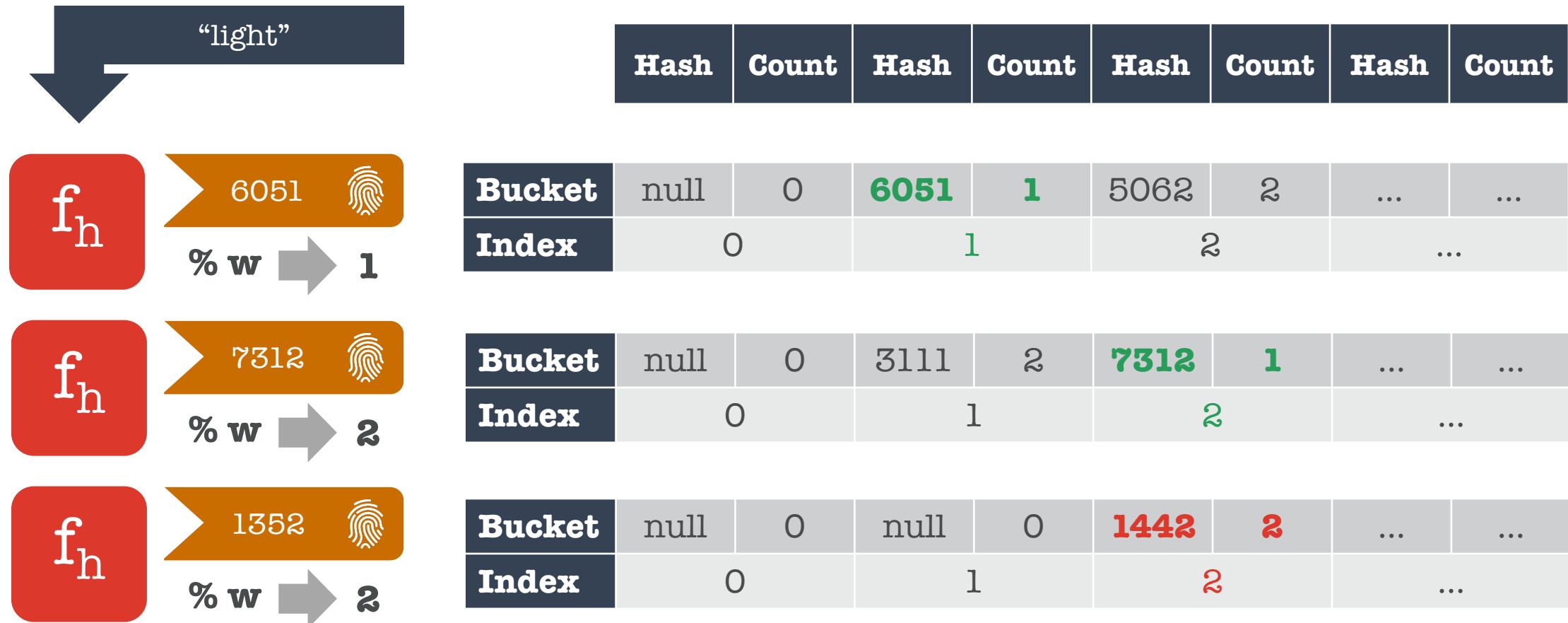
Hash	Count	Hash	Count	Hash	Count	Hash	Count
------	-------	------	-------	------	-------	------	-------

Bucket	null	0	null	0	<b>5062</b>	<b>2</b>	...	...
Index	0		1		<b>2</b>		...	...

Bucket	null	0	<b>3111</b>	<b>2</b>	null	0	...	...
Index	0		<b>1</b>		<b>2</b>		...	...

Bucket	null	0	null	0	<b>1442</b>	<b>2</b>	...	...
Index	0		1		<b>2</b>		...	...

# Adding a Different Value



# Probability of Decay

$$p = b^{-c}$$

$$p = 1.05^{-2}$$

$$p = 0.907$$

- $p \rightarrow$  probability of decay
- $b \rightarrow$  rate of decay
  - $-b > 1$
  - $-b \approx 1$
- $c \rightarrow$  the count

# Decay Has Occurred

↓  
“light”

**f<sub>h</sub>** → 6051   
% w → 1

**f<sub>h</sub>** → 7312   
% w → 2

**f<sub>h</sub>** → 1352   
% w → 2

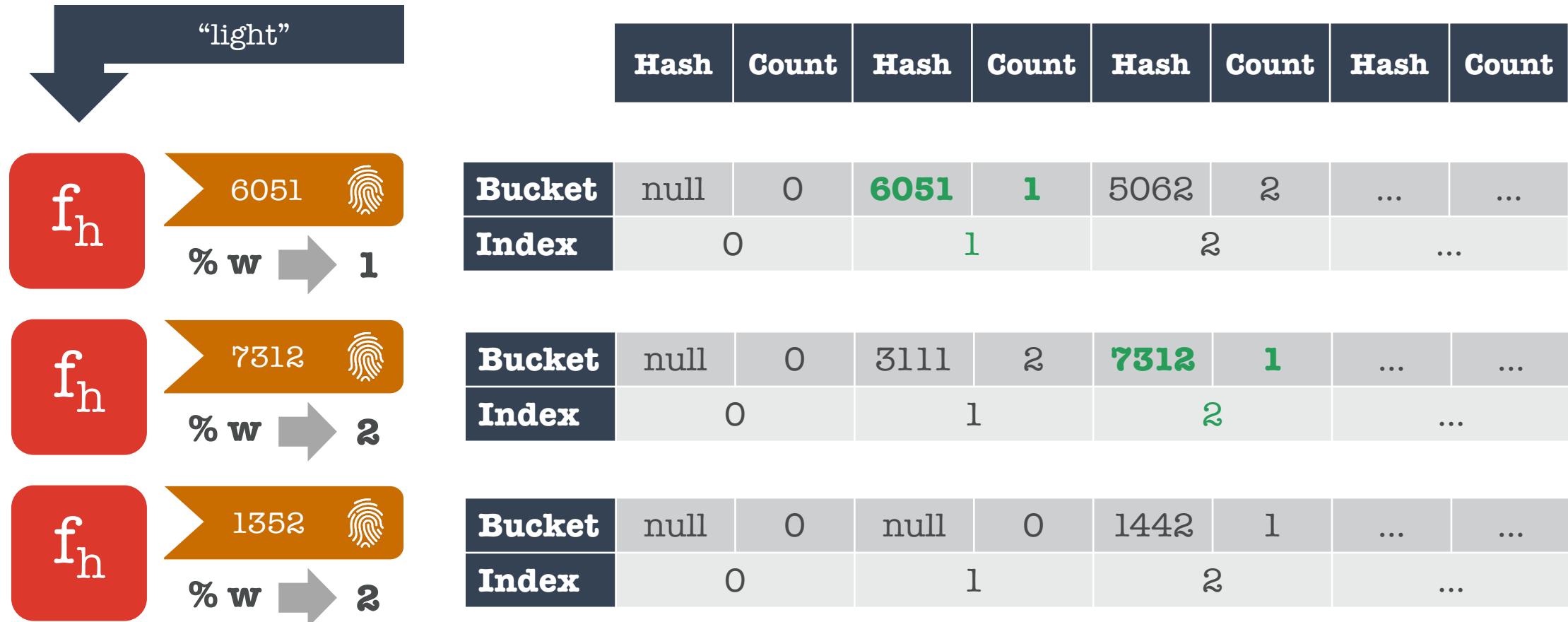
Hash	Count	Hash	Count	Hash	Count	Hash	Count
------	-------	------	-------	------	-------	------	-------

Bucket	null	0	6051	1	5062	2	...	...
Index	0		1		2		...	...

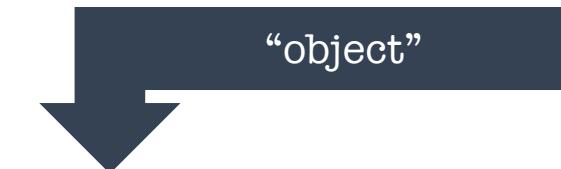
Bucket	null	0	3111	2	7312	1	...	...
Index	0		1		2		...	...

Bucket	null	0	null	0	<b>1442</b>	<b>1</b>	...	...
Index	0		1		<b>2</b>		...	...

# Querying After Decay



# Querying the Count



Bucket	null	0	null	0	<b>5062</b>	<b>2</b>	...	...
Index	0		1		<b>2</b>		...	...

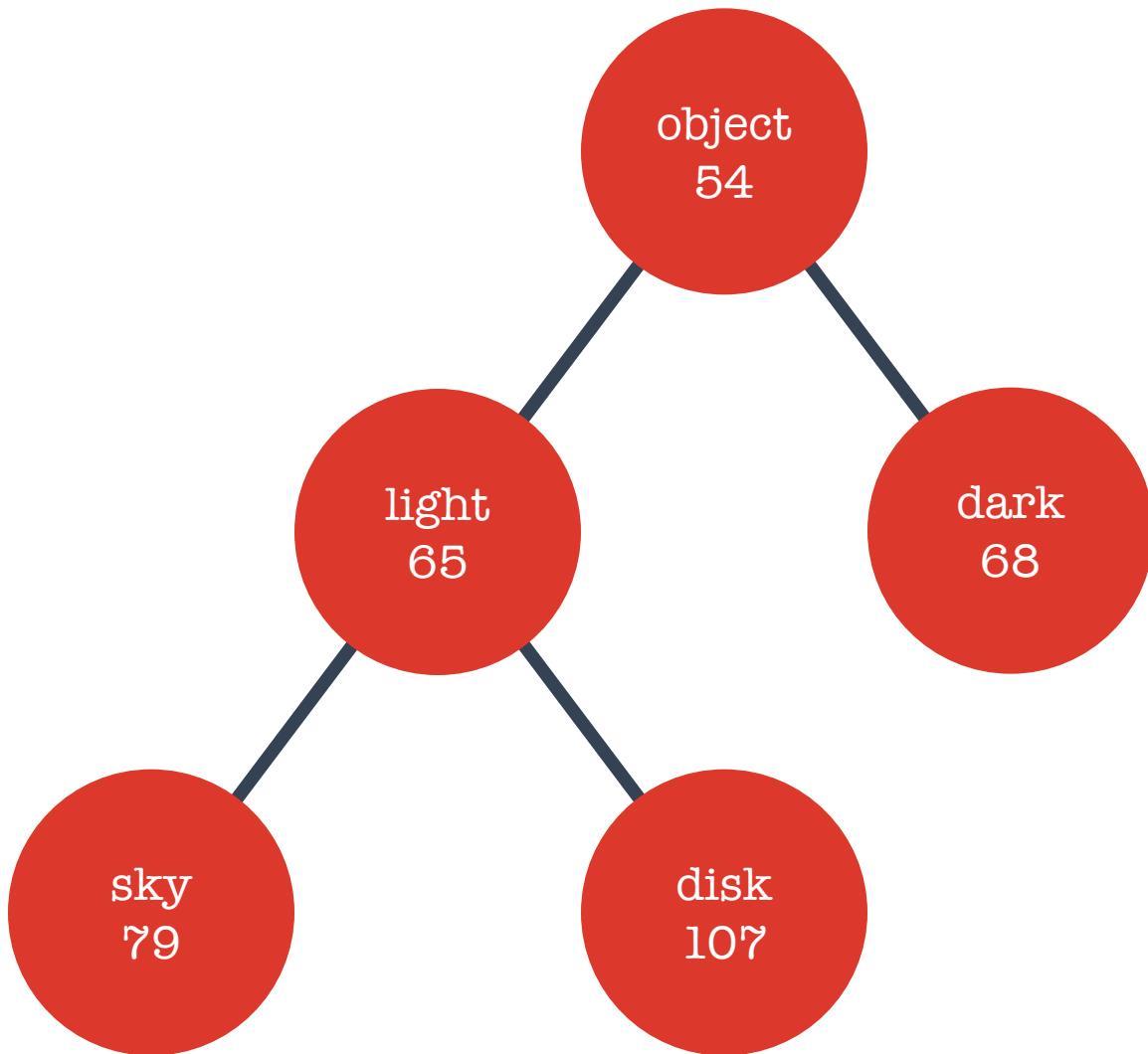


Bucket	null	0	<b>3111</b>	<b>2</b>	null	0	...	...
Index	0		1		2		...	...



Bucket	null	0	null	0	<b>1442</b>	<b>1</b>	...	...
Index	0		1		<b>2</b>		...	...

# Updates the MinHeap Every Time



# Using TopK in Redis

>	TOPK.RESERVE	ufo:shapes	3	100	5	0.9
	command	key	number	width	depth	decay
OK						

# Adding to TopK in Redis

>	TOPK.ADD	ufo:shapes	disk	
	command	key	Item to add	
(nil)				

>	TOPK.ADD	ufo:shapes	disk	teardrop
	command	key	Item to add	Another item to add
(nil)				

>	TOPK.ADD	ufo:shapes	light	disk	teardrop
	command	key	Item to add	Another item to add	Another item to add
(nil)					

# Using TopK in Redis

>	TOPK.QUERY	ufo:shapes	disk
	command	key	Item to add
(integer) 1			

>	TOPK.COUNT	ufo:shapes	teardrop
	command	key	Item to add
(integer) 2			

>	TOPK.LIST	ufo:shapes	
	command	key	
light teardrop disk			

A dark, atmospheric night scene. In the center, a person with long hair and a backpack stands on a paved path, looking up at a bright street lamp. The lamp is mounted on a tall pole and casts a strong, glowing light that illuminates the surrounding trees and foliage. The sky is dark and filled with mist or fog. The overall mood is mysterious and contemplative.

Demo

# Resources



## Bloom Filters by Example

<https://llimllib.github.io/bloomfilter-tutorial/>

## Bloom Filter Calculator

<https://hur.st/bloomfilter/>

## MinHash Tutorial

<https://mccormickml.com/2015/06/12/minhash-tutorial-with-python-code/>

## HeavyKeeper: An Accurate Algorithm for Finding Top-k Elephant Flows

<https://www.usenix.org/conference/atc18/presentation/gong>

## UFO Sightings Dataset

<https://data.world/timothyrenner/ufo-sightings>

## The National UFO Reporting Center

<http://www.nuforc.org/>

## Redis Bloom

<http://redisbloom.io/>

## Meet Top-K

<https://redislabs.com/blog/meet-top-k-awesome-probabilistic-addition-redisbloom/>

[https://github.com/guyroyse/  
understanding-probabilistic-data-  
structures](https://github.com/guyroyse/understanding-probabilistic-data-structures)



**redis**labs  
HOME OF REDIS

# Guy Royse

## Developer Advocate

## Redis Labs

 @guyroyse

 [github.com/guyroyse](https://github.com/guyroyse)

 [guy.dev](https://guy.dev)



A dark, atmospheric night scene. In the center, a person stands on a paved path, looking up at a bright street lamp. The lamp's light illuminates the surrounding trees and creates a hazy glow. The overall mood is mysterious and contemplative.

Thanks!



**redis**labs  
HOME OF REDIS